

AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method for maintaining service dependency relationships between executable predefined service components in a computer system comprising the steps of:

maintaining a dynamic service consistency file containing entries to identify predefined service components that are currently available in the computer system, wherein the currently available predefined service component entries are linked according to their dependency, the dynamic service consistency file comprising a first representation in any one from the group of a tree, a graph, a linked list, or a table;

comparing the dynamic service consistency file to a reference file containing entries to identify the predefined service components in the computer system, wherein the predefined service component entries are linked in the reference file according to their dependency, the reference file comprising a second representation in any one from the group of a tree, a graph, a linked list, or a table;

determining whether an inconsistency exists between service component entries within the dynamic service consistency file and the reference file; and

starting any lost predefined service component to correct any inconsistency based upon the determining step.

2. (Original) The method of claim 1 further comprising the steps of:

modifying the dynamic service consistency file based upon the starting step; and
repeating from the maintaining step.

3. (Original) The method of claim 1 further comprising the steps of:

reading the reference file;
identifying the dependency services of the lost predefined service component according to the reference file; and

generating a log message to report the lost predefined service component including the identified dependency service components based upon the identifying step.

4. (Original) The method of claim 3 further comprising the step of saving the log message.
5. (Original) The method of claim 1 further comprising the steps of:
reading the reference file;
identifying the dependency services of the lost predefined service component according to the reference file;
determining whether the lost predefined service component has been successfully started;
generating a log message to report the lost predefined service component according to the determination step, wherein the log message includes dependency service information of the lost predefined service component.
6. (Original) The method of claim 5 further comprising the steps of:
generating an alert message to report the lost predefined service according to the determination step; and
sending the alert message to a user.
7. (Original) The method of claim 1 wherein the maintaining step further comprises the steps of:
identifying the currently available predefined service components; and
generating the dynamic service consistency file based upon the identifying step.
8. (Original) The method of claim 7 wherein the maintaining step further comprises the steps of:
determining whether a timeout has occurred; and
repeating the identifying step when a timeout has occurred based upon the determination step.
9. (Original) The method of claim 1 wherein inconsistency is based on at least one missing predefined service component in the dynamic service consistency file when compared to the reference file.

10. (Original) The method of claim 1 wherein the service components include any one from the group of a process, a service hosting process, a service, a provider service, and a dependency service.

11. (Original) The method of claim 1 wherein the predefined service components are installed service hosting processes or provider services that are depended upon by other services.

12. (Canceled)

13. (Canceled)

14. (Withdrawn) A method for generating a reference file including dependency links of executable installed service components in a computer system, comprising the steps of:

reading configuration information from a registry;
identifying dependency links of the installed service components based upon the reading step; and
generating the reference file based upon the identifying step, wherein the reference file contains entries to identify the installed service components linked according to their dependency.

15. (Withdrawn) The method according to claim 14 further comprising the steps of:
determining whether there is a change in configuration of the installed service components;

identifying the change in the installed service components based upon the determination step; and

updating the reference file based upon the identification step.

16. (Withdrawn) A method for creating a file to indicate service dependency relationships between executable service components in a computer system, comprising:

creating a plurality of nodes to identify the service components;
wherein dependencies of the service components are linked.

17. (Withdrawn) The method of claim 16, wherein a data structure of the file is any one from the group of a tree, a graph, linked list, or a table.

18. (Withdrawn) The method of claim 16, wherein the file is a reference file containing entries to identify installed service components in the computer system or a dynamic service consistency file containing entries to identify currently available service components in the computer system.

19. (Currently Amended) A computer system for maintaining service dependency relationships between executable service components in a computer system, comprising:

a reference file containing entries to identify installed service components in the computer system, wherein the installed service component entries are linked according to their dependency, the reference file comprising a second representation in any one from the group of a tree, a graph, a linked list, or a table;

a dynamic service consistency file containing entries to identify currently available service components in the computer system, wherein the currently available service component entries are linked according to their dependency, the dynamic service consistency file comprising a first representation in any one from the group of a tree, a graph, a linked list, or a table; and

a server control manager for comparing the dynamic service consistency file to the reference file to identify any inconsistency between the files, and starting any lost service component to correct any identified inconsistency.

20. (Currently Amended) A computer-readable medium having computer-executable instructions for performing steps comprising:

maintaining a dynamic service consistency file containing entries to identify predefined service components that are currently available in ~~the~~ a computer system, wherein the currently available predefined service component entries are linked according to their dependency, the dynamic service consistency file comprising a first representation in any one from the group of a tree, a graph, a linked list, or a table;

comparing the dynamic service consistency file to a reference file containing entries to identify the predefined service components in the computer system, wherein the predefined

service component entries are linked in the reference file according to their dependency, the reference file comprising a second representation in any one from the group of a tree, a graph, a linked list, or a table;

determining whether an inconsistency exists between service component entries within the dynamic service consistency file and the reference file; and

starting any lost predefined service component to correct any inconsistency based upon the determining step.

21. (Withdrawn) A computer-readable medium having computer-executable instructions for performing steps comprising:

reading configuration information from a registry;
identifying dependency links of installed service components based upon the reading step; and

generating a reference file based upon the identifying step, wherein the reference file containing entries to identify the installed service components linked according to their dependency.

22. (Withdrawn) A computer-readable medium having computer-executable instructions for performing steps comprising:

creating a plurality of nodes to identify a plurality of service components;
wherein dependencies of the service components are linked.